- Attribute Assistant
  - The Attribute Assistant is an ArcMap extension that provides advance abilities to calculate attribute data. The extension is driven by two configuration tables, DynamicValues and GenerateID. These tables define the rules and the values for the extension.
  - The Extension also includes four ArcMap Commands and one ArcMap toolbar – **Attribute Assistant Toolbar**.
    -  **Attribute Assistant On/Off**
      - **Overview:** This tool toggles the Attribute Assistant on and off. It will also turn red and deactivate if some of the required tables are missing.
      - **Note:** This tool are not installed with the Infrastructure Editing Template, it is part of the Attribute Assistant.
      - **Requirements**: Attribute Assistant Add-In installed.
    -  **Run Change Rules for Selected Features**
      - **Overview:** This tool triggers a change event on each selected feature which if the Attribute Assistant is on, process all the rules defined in the Dynamic Value Table.
      - **Note:** This tool are not installed with the Infrastructure Editing Template, it is part of the Attribute Assistant.
      - **Requirements**: Attribute Assistant Add-In installed, turned on and have a set of selected features.
    -  **Run Manual Rules for Selected Features**
      - **Overview:** This tool triggers a manual event on each selected feature which if the Attribute Assistant is on, process all the rules defined in the Dynamic Value Table.
      - **Note:** This tool are not installed with the Infrastructure Editing Template, it is part of the Attribute Assistant.
      - **Requirements**: Attribute Assistant Add-In installed, turned on and have a set of selected features.
    -  **Load Last Value Array**
      - **Overview:** This tool loops through the Dynamic Value table and present a dialog for each LAST_VALUE field. You can see the existing value and set a new one
      - **Requirements**: Attribute Assistant Add-In installed and the dynamic value table populated.
      -
  - The Dynamic Value table consist of 6 fields that define the rules and how that interact with your features
    - **Note**: This table can be versioned.

- **Note**: You can use multiple DynamicValue tables, but only one can be present in ArcMap at a time. The database table name can be anything you would like, but the name in the TOC must match the name listed in the config, which by default is DynamicValue. The code is designed to check the TOC name first, then the Database name.

Table Name	Field Name	Value Method	Value Info	Create	Change
*	LASTEDITOR	CURRENT_USER	W	True	True
ssGravityMain	FACILITYID	EXPRESSION	replace([FROMMH] & "-" & [TOMH]),"MH-	False	True
ssGravityMain	SLOPE	EXPRESSION	(((Abs([UPELEV]-[DOWNELEV]))/[Shape	True	True
SpecialEvent	EVENTID	EXPRESSION	("EV-"& [OBJECTID])	True	False
wPump	INLETDIAM	FROM_EDGE_FIELD	DIAMETER	True	True

- **Table Name:** This field defines the layer the rule will be ran on. You can use * for all layers.
  - **Note:** You can specify a subtype in the table by using the following format:
    - **FeatureClassName|Subtype values|Invalid Subtypes**
    - Make sure to use the code, not the description.
    - This gives you the option to process this rule on a layer with all subtypes(*) but n,m,x
    - Ex: wFittings|*|2,3 - All subtypes are processed but 2 and 3
    - Ex: wFittings|1,5,6 - Only fittings with subtypes of 1,5,6 are processed
- **Field Name:** The name of the field to store the result of the rule. Rules that require null or ignore the Field Name field
  - **CREATE_LINKED_RECORD**
  - **UPDATE_INTERSECTING_FEATURE**
  - **FROM_EDGE_MULTI_FIELD_INTERSECT**
  - **TO_EDGE_MULTI_FIELD_INTERSECT**
  - **MULTI_FIELD_INTERSECT**
  - **NEAREST_FEATURE_ATTRIBUTES**
  - **SPLIT_INTERSECTING_FEATURE**

- **Value Method:** The rule to run.
- **Value Info:** The configuration of the rule. Not all rules require a Value Info value.
- **Create:** If True, the rule will run when a feature is created.
- **Change:** If True, the rule will run when a feature is changed.
- **Manual:** If True, the rule will run when a run Manual button is clicked.

- The Generate ID table consist of two fields with many rows. Each row represents a sequence name and a sequence.
  - ***Warning***: This table cannot be versioned. If versioned, duplicate ID's will be generated.

OBJECTID *	Sequence Name	Sequence Counter
2	Test1	6
3	Counter	4
4	Test1ID	5
5	Test1MO	9

- **Optional Debug Mode:** There is an optional Debug mode that can be turned on in the configuration file. Enabling this mode will write out each step to a log file and will help you locate any errors or issues with the extensions. When you start editing, you will be prompted for the log file location.
- **Configuration:** There a few configuration options since the two tables provide the bulk of the configuration.

```
<!--Options for the Attribute Assistant-->
<!--Option to turn on the AA on startup-->
<add key="AttributeAssistant_EnabledOnStartUp" value="false" />
<!--The name of the table that defines the functions for
  the Attribute Assistant-->
<add key="AttributeAssistant_TableName" value="DynamicValue" />
<!--The table used for Unquie ID Generation-->
<add key="AttributeAssistant_GenerateId_TableName" value="GenerateId" />
<!--Activate the debug mode which writes a log file-->
<add key="AttributeAssistant_Debug" value="True" />
<!--Option to clear the last values between toggling it on and off-->
<add key="AttributeAssistant_ClearLastValue" value="False" />
```

Below is a table of the rules that you can configure for your data

Value Method	Value Info	Details	Requirements
ANGLE	Value Info: **A or G**	This rules store the angle of a line into the field specified in Field Name. You can get the angle either as Geographic or Arithmetic.	**FieldName:** Field to store the Angle value

AUTONUMBER	**No Values, just list the field to autonumber in the field name**	This rules will find the largest number in an integer field and store the next number in the field specified	
CASCADE_ATTRIBUTE	**Table or Layer|Field|Suppress Prompt** **Ex: Address|Name|False**	This rules will monitors a field and when an edit is made to that field, it searches all specified layers for the original value and makes the change in those layers. By default, it will prompt the user the verify that they want to push the changes, you can suppress it by entering F or False for the third option.	
COPY_LINKED_RECORD	**Feature Layer|Field To Copy|Primary Key Field|Foreign Key Field**	This rules copies a value from a related table into the source feature based on a primary/foreign key relationship	**FieldName:** Field to store the value
COPY_FEATURE	**Value Info: Field Value|Target Feature Layer|Edit Tempalte to use|Cut or Copy|Field Pairs(format: Source Field-Target Field,…..)** **Example: Field Name: ACTIVEFLAG** **Value Info: 0|wAbandonedLine|Abandoned Water Mains|CUT|DIAMETER-DIAMETER,MATERIAL-MATERIAL** **Or** **Value Info: 0|wAbandonedLine** **Or** **Value Info: 0|wAbandonedLine|||**	This rules will monitor a field and when the field is changed and it matches the value specified in the value info, it will copy or cut the feature and place it in the specified target layer. It will copy field names that match, but you also have the option of listing field pairs in the value info. You can specify a Feature Template to use for the target. The feature template, copy or cut(copy is default), and field pairs can be blank. Field Name: Specify the field to monitor for rule to be value	

CREATE_LINKED_RECORD	**Feature Layer|Field To Copy|Field To Populate|Primary Key Field|Foreign Key Field** **Example:** **wLinkedRecords|MapSheetID|MapSheet|FacilityID|ForeignKey**	Creates a new record in a table and creates a relationship to table using the primary key in the edited record.	The Field Name Field must be blank or null for this rule to work
CREATE_PERP_LINE	**Value Info: Layer to Search for|Offset Value or Field with offset value|Search Distance|Snap To line|Target Line Layer|Optional Target Edit Template Value Info** **Ex: sewpipes|5|5|True|com_pipes|com_pipes**	This rules is similar to offset, but instead of storing the XY offset coordinates, it creates a line to the offset location	**FieldName**: Empty
CURRENT_USER	**W** - stores full windows login name as domain\username **U** - stores just the windows username **D** - stores the connected database user for the edit session If you leave VALUE_DATA blank, it will store the database user name if available otherwise store full windows login.	Stores current user name.	String Field
EDGE_STATS	Field to analyze | Type of Analysis **Valid Types:** o **Min** o **Max** o **Sum** o **Mean**	This rule uses the geometric network to find all connected edges and provides stats on a field in all connected features. Concat put all the values in the field delimited by a ",". Only works if the FieldName Field is a text field.	**Defined on a Geometric network Junction Feature Class**

	o **Concat** **Example: DIAMETER|Max**		
Example: SERVER=millerlaptop_sqlexpress;INSTANCE=sde:sqlserver:millerlaptop\sqlexpress;DATABASE=LocalGovernment;AUTHENTICATION_MODE=OSA;VERSION=dbo.DEFAULT|LocatorName SERVER=millerlaptop_sqlexpress;INSTANCE=sde:sqlserver:millerlaptop\sqlexpress;DATABASE=LocalGovernment;USER=User;PASSWORD=Pass;VERSION=dbo.DEFAULT|LocatorName millerlaptop_sqlexpress;sde:sqlserver:millerlaptop\sqlexpress;LocalGovernment;OSA;dbo.DEFAULT|LocatorName millerlaptop_sqlexpress;sde:sqlserver:millerlaptop\sqlexpress;LocalGovernment;username;password;dbo.DEFAULT|LocatorName C:\Projects\LocalGovernmentResourceCenter\Maps and GDBs\Locators|LocatorName C:\Projects\LocalGovernmentResourceCenter\Maps and GDBs\Locators\Address.gdb|LocatorName			
EXPRESSION	**Expression** **Example**: replace(([FROMMH] & "-" & [TOMH]),"MH-","") Iif([DIAM] <= 6, "RED", Iif ([DIAM] <=10 , "ORANGE", Iif ([DIAM] <=15 , "GREEN","BLUE")))	Stores the results of an expression. You can use IIF statements to create complex If Else Statements. If you want to run the expression on the edited field, use #, in the Field Name. In the Expression, you can list the edited field using [#].	
FEATURE_STATS	Field to analyze | Type of Analysis **Valid Types:**	This tool evaluates the list of fields and provides stats on those fields. Concat put all the values in the field delimited by a ",". Only works if the FieldName Field is a text field.	

	o **Min** o **Max** o **Sum** o **Mean** o **Concat** **Example:** **ssGravityMain|DIAMETER|Max|25**		
FIELD	**Field Name|CODE or DISPLAY**	Copies one field to another field in the same feature. If you leave off the second argument, the display value will be used for Subtypes or Domains.	
FIELD_TRIGGER	**Field Value|Field to Change|Value to Set** **Ex:** **ABANDONED|PRESSUREZONE|NONE**	This rule monitors a field and when it changes, checks the value to the FieldValue in the Value Info. If they match, it updates the listed field with the Value to Set.	**FieldName:** Field to Monitor for Change
FROM_EDGE_FIELD	**Field to Copy|FeatureClass Name copy from|Field to Check|Value in field** **Ex:** **FACILITYID|wServiceConnection|Service Type|Domestic**	Transfers a field value from a connected edge feature to a junction feature. The feature can be restricted f you fill out the last three options, list the FC name, the field and the value to match.	Must be assigned to a point feature class that participates in a geometric network. Requires ArcEditor or ArcInfo.
FROM_EDGE_MULTI_FIELD_INTERSECT	Field to copy | Array of fields to populate **Example:** **DIAMETER|Diam1,Diam2,Diam3**	This rules uses the geometric network to find all **From** edges from a junction and stores values from each intersecting feature into a series of fields in the source layer.	**Field Name:** null or blank **Defined on a Geometric network Junction Feature Class**
FROM_EDGE_STATS	Field to analyze | Type of Analysis	This rule uses the geometric network to find the **From** edges and provides stats on a field in all connected features. Concat put all the values in the field delimited by a ",". Only works if the FieldName	**Defined on a Geometric network**

	Valid Types: o **Min** o **Max** o **Sum** o **Mean** o **Concat** **Example: DIAMETER|Max**	Field is a text field.	**Junction Feature Class**
FROM_JUNCTION_FIELD	**Field to Copy|FeatureClass Name copy from|Field to Check|Value in field** **Ex:** **FACILITYID|wServiceConnection|Service Type|Domestic**	Stores a value that is obtained from a specified field in the junction feature at the start of the currently edited line. The feature can be restricted f you fill out the last three options, list the FC name, the field and the value to match.	Must be assigned to a line feature class that participates in a geometric network. Requires ArcEditor or ArcInfo.
GENERATE_ID	**<SequenceName>| <optionalSequence Width>|<optionalFormatString>[seq]** **Example:** WHYDRANT|0|HYD-[seq]	Increments a row in an unversioned table and stores that newly incremented value. Uses an unversioned table that is typically called GenerateId (change in AttributeAssistant.config) to store and increment unique sequence numbers.	
GENERATE_ID_BY_INTERSECT	<Layer Name> |<Layer Field Name>|<Sequence Field Name from GenerateID Table>| |optionalSequenceWidth|[id]optionalFormatString[seq]<any order of ID and SEQ> **Example**: Grid|GRID_ID|GRIDIDX|4|GRID[ID]-[SEQ]	The result would look like GRID5-0002, if the grid you intersected ID was 5 and the next number in the table was 2. This tool requires you create fields in the Generate ID table that are a combination of the GRIDIDX(the Sequence Field Name) and the Grid ID. So the tools looks at the intersected grid, gets the ID from it, combines it with the <Sequence Field Name from GenerateID Table> to look for a field in the Generate ID table to get an seq for that grid. Say the grid you created a feature in was 5, you need a field called GRIDIDX5 in the generate ID table.	
GET_ADDRESS_FROM_CENTERLINE	**Value Info:** Road Feaure Class Name|Road Name field, Left To Field, Right To	This rule will find the closest point on a road and extract the address information from that location. Very similar to a reverse geocode,	Note, this tool always search the feature

	Field, Left From Field, Right From Field | Optional Search distance	but a locator service is not used	class, not the layer.
GET_ADDRESS_USING_ARCGIS_SERVICE	**Url** to a Geocoding service|WKID **Example:** http://myServer/arcgis/rest/services/Address_with_Zone|4326	This method preforms a reverse geocode. The default service is the ArcGIS.com geocoding service. You can specify your own service. If the coordinate you are passing to the service is in a difference coordinate system, you need to specify the projection of the service, so the location can be projected.	String Field
GET_ADDRESS_USING_GEOCODER	Path to Locator and Locator Name	This method preforms a reverse geocode.	String Field
GUID	Optionally enter one of the letters below to format the GUID as desired. **N** - GUID with no special characters - length 32 **D** - GUID with dashes - length 36 **B** - GUID with dashes and braces - length 38 **P** - GUID with dashes and parenthesis - length 38 **default** - GUID with dashes and braces - length 38 Leave the ValueInfo Field blank to get the default GUID format.	Stores a new GUID.	The target field must be a field type string field and must be long enough to store the desired format of GUID.
INTERSECT_STATS	Feature Layer to search for | Field to analyze | Type of Analysis | Search Tolerance **Valid Types:**	This rule search for intersecting features and provide stats on a field in all connected features. Concat put all the values in the field delimited by a ",". Only works if the FieldName Field is a text field.	

	o **Min** o **Max** o **Sum** o **Mean** o **Concat** **Example: ssGravityMain|DIAMETER|Max|25**		
INTERSECTING_EDGE	**Field To copy info from**	The rules looks for a field value in the first intersecting edge feature. This rule is helpful if you want to transfer a value from a geometric network feature if it moves. For example, you move a hydrant and you want to copy a value from the connected lateral. The move of the hydrant completes before the network moves the lateral, so Intersecting_Feature will not find the lateral.	
INTERSECTING_FEATURE	**<Layer Name>,<Layer Name>,..|<Field Name>|C or P or F** **Example:** ssPressurizedMain,ssGravityMain|DIAMETER|C	Gets a value from an intersecting feature in the specified layer. You can specify any number of layers to look for by listing them with commas between their names. Third option lets you change how you search or the results are returned. Options are, P or Prompt, C or Centroid, or F or First.	
INTERSECTING_FEATURE_DISTANCE	**<Layer Name>, <Layer Name>,..|<Field Name>** **Example:** ssPressurizedMain,ssGravityMain|FACILITYID	Gets a value from an intersecting feature in the specified layer and reports the distance along the line. You can specify any number of layers to look for by listing them with commas between their names.	
INTERSECTING_LAYER_DETAILS	**RASTER OR VECTOR OR BOTH or List of Layers|P or N|C or P or F|Match pattern**	Extracts the details from a Layer. This tool can be used to get the name or path the layer. Specify P will extract the full path, where N will extract just the name. Third option lets you change how you search or the results are returned. Options are, P or Prompt, C or	

	Example: Layer1, Layer2,….|P|C RASTER|P|C|*.png VECTOR|N|P	Centroid, or F or First. The Match pattern lets you limit your search to layers that contain that pattern.	
INTERSECTING_RASTER	**<Layer Name>,<Layer Name>,..|Label** **Example:** FiveMeterSurface|Elevation:	Gets a value from an intersecting raster cell in the specified layer. You can specify any number of layers to look for by listing them with commas between their names.	
JUNCTION_ROTATION	**A or G|Additional spin|Special Field** **A** - if you enter an A it will store the rotation using an arithmetic rotation. If you leave VALUE_DATA blank or use **G**, it will store the rotation using a geographic rotation. Example: **G|90|Diameter**	Stores a rotation angle for a junction feature based on connected edge features by storing a rotation angle in the specified field. If you specify a Special field, the tool will look at a values in the field, must be numeric, and spin the feature from the smaller to larger value.	
LAST_VALUE	Option to not change values no-null values|Default Value Ex: True|PVC	Repeats the last value used for a field.	
LATITUDE	None	Stores the Y coordinate projected to WGS84 decimal degrees, the centroid is used on a line or polygon	
LENGTH	None	Stores calculated length of line feature.	

LINK_TABLE_ASSET	**Feature Layer| Field to copy** **Example:** wMain|FacilityID	Updates a field in the table or layer with a value from a selected feature. You can list multiple fields in the FieldName box. The rule will loop through the selected features and copy the one field out of each feature into the list of fields in the order of the fields in the FieldName.	
LONGITUDE	None	Stores the X coordinate projected to WGS84 decimal degrees, the centroid is used on a line or polygon	
MINIMUM_LENGTH	Minimum length of line	This rule will find the nearest feature to the source feature and copy a series of attributes	
MULTI_FIELD_INTERSECT	Feature Layer to search for | Field to copy | Array of fields to populate|Search Tolerance **Example:** **ssGravityMain|DIAMETER|Diam1,Diam2,Diam3|25**	This rules searches for intersecting features in a new layers and stores values from each intersecting feature into a series of fields in the source layer.	**Field Name:** null or blank
NEAREST_FEATURE	**<Layer Name>,<Layer Name>..|<Field Name>|<Search Distance>|<Optional>SourceFieldToMatch|<Optional>TargetFieldToMatch** **Example:** wMeter,sLateralPoint|ACCOUNT|100 or wMeter,sLateralPoint|ACCOUNT|100|DIAMETER|DIAMETER	Gets a value from the nearest feature in the specified layer. You can optionally list a field in the layer being modified and the layers to search for. Only values that match in these fields will be honored.	
NEAREST_FEATURE_ATTRIBUTES	**Feature Class Search|Fields in that**	This rule will find the nearest feature to the source feature and copy	**Separate multi**

	feature class|Fields in the source Featureclass| Distance. **Example: wServiceConnection|LOCATIONID,LASTEDITOR|SVCLOCID,SVCEDITOR|50**	a series of attributes	**fields by a ",". The number of fields must match.** **Field Name:** null or blank
OFFSET	Value Info: **Layer to Search for|Offset Value or Field with offset value|Search Distance Template** Value Info Ex: **sewpipes|5|5**	This rules uses the input point to find the closest line and use that intersection to construct and offset point perpendicular to the line. The X and Y coordinates are stored in the fields specified in the Field Name. **FieldName**: Two fields, separated by a comma, to store the x and y values in **FieldName Ex:** X_Offset_Coord, Y_Offset_Coord	
PROMPT	Blank or null	This field looks at a list of fields in FieldName field and prompts the user for input if no value exist. The fields in FieldName are comma delimited. If the field is a subtype of domain field, those options are presented in the dialog for the user to select.	
SET_MEASURES	If you leave ValueInfo blank, it will calculate the Ms starting with zero and ending with the length of the line. If you enter a **P** for ValueInfo, it will calculate the Ms starting with zero and ending with 100.	Populates the M coordinates in a line which enables using Add Route Events to point and line events dynamically along line features	Requires a line with M's turned on
SIDE	**Line Layer|Line ID Field|ID in Input** **ssGravityMain|OBJECTID|MainID**	This rule uses an ID from the input point layer and find matching ID in the listed line layer and determines the side of the line the point is on.	Point input and Line layer listed in the Value Info
SPLIT_INTERSECTING_FEATURE	Feature Class to split search for to split when source feature has been created or modified, this can be a list of layers,	This rule will attempt to split the layer specified in the Value info if they intersect	**Field Name:** null or blank

	delimited by the | character. **Example: wConstructionLine|wMain**		
TIMESTAMP	None	Stores current date and time.	Date or String Field
TO_EDGE_FIELD	**Field to Copy|FeatureClass Name copy from|Field to Check|Value in field** **Ex:** **FACILITYID|wServiceConnection|Service Type|Domestic**	Transfers a field value from a connected edge feature to a junction feature. The feature can be restricted f you fill out the last three options, list the FC name, the field and the value to match.	Must be assigned to a point feature class that participates in a geometric network. Requires ArcEditor or ArcInfo.
TO_EDGE_MULTI_FIELD_INTERSECT	Field to copy | Array of fields to populate **Example:** **DIAMETER|Diam1,Diam2,Diam3**	This rules uses the geometric network to find all **To** edges from a junction and stores values from each intersecting feature into a series of fields in the source layer.	**Field Name:** null or blank **Defined on a Geometric network Junction Feature Class**
TO_EDGE_STATS	Field to analyze | Type of Analysis **Valid Types:** o **Min** o **Max** o **Sum** o **Mean** o **Concat** **Example: DIAMETER|Max**	This rule uses the geometric network to find the **To** edges and provides stats on a field in all connected features. Concat put all the values in the field delimited by a ",". Only works if the FieldName Field is a text field.	**Defined on a Geometric network Junction Feature Class**
TO_JUNCTION_FIELD	**Field to Copy|FeatureClass Name copy**	Stores a value that is obtained from a specified field in the junction	Must be assigned to

	from|Field to Check|Value in field **Ex:** **FACILITYID|wServiceConnection|Service Type|Domestic**	feature at the end of the currently edited line. The feature can be restricted f you fill out the last three options, list the FC name, the field and the value to match.	a line feature class that participates in a geometric network. Requires ArcEditor or ArcInfo.
UPDATE_INTERSECTING_FEATURE	**<Intersecting Layer|<Field in Intersecting Layer>|Value or <Field in edited layer>**	Updates a field in an intersecting feature with a value or a field value from the modified or created feature	The Field Name Field must be blank for this rule to work
VALIDATE_ATTRIBUTE_LOOKUP	**Table or Layer|Field**	This rules will monitors a fields value and checks that value against a lookup table to make sure it is valid. The tool uses wildcard searching, so you can specify part of the value you are looking for. If multiple matches are found, a prompt will be present to the user to select a valid value.	
VALIDATE_ATTRIBUTES	Field1,Field2,Field3,.. **Example: DIAMETER,MATERIAL,Line Type**	This rules compares the values in the fields list to all feature templates for the feature class. If a template is not found with matching values, the edit is cancelled. Null values are skipped. If you list a field in the Field Name, this rule will only be checked when that field is changed, you can list multiple fields, separated by a ",".	
VALIDATE_CONNECTIVITY	List the number of valid connections. If you specify a field in Field name you can list different values for that field. Example 1: No field defined and 1,2 and 4 connections are valid 1|2|4 Example 2: No field defined and a bend with 1, 3 connections is valid and a tee with 1,2,3,4, connections are valid	This tools checks the number of connections on a feature and rejects the edits if the criteria is not meet.	

	Bend,1| Bend,3|Tap,1,4		
X_COORDINATE	None	Stores the X coordinate in database units, the centroid is used on a line or polygon	
Y_COORDINATE	None	Stores the Y coordinate in database units, the centroid is used on a line or polygon	